Remarks

I. Claim Rejections – 35 U.S.C. §103

A. The Office Action rejects claims 1- 6 and 8-20 under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 6,233,242 to Mayer et al. (hereinafter "Mayer") in view of U.S. Patent No. 5,682,534 to Kapoor et al. (hereinafter "Kapoor"). Regarding claim 1, the Office Action states:

> As per claim 1, Mayer teaches a method comprising:
>
> maintaining a set of communication control blocks (CCBs), some of the set of CCBs being maintained in a static random access memory (SRAM), others of the set of CCBs being maintained in a dynamic random access memory (DRAM), wherein a first plurality of the set of CCBs is under control of a network interface device, and wherein a second plurality of the set of CCBs is under control of a processing device, the processing device being coupled to the network interface deice, the processing device executing a network protocol stack (e.g. Mayer, several types of "control blocks" are disclosed, high speed bus control block, memory control block, processor control block, all relating to communication procedures also the maintaining of the control blocks in both DRAM and SRAM, col. 17, lines 11-17; col. 31, lines 63-67; col. 32, lines 1-30);
>
> receiving a packet onto the network interface device from a network, the packet including a data portion and a header portion (e.g. Mayer, col. 7, lines 52-67);
>
> using content addressable memory (CAM) on the network interface device to determine that the packet is associated with one of the first plurality of CCBs (e.g. Mayer, CAM is taught as an alternative form of SRAM, similar in both use and function, col. 2, lines 28-35; col. 32, lines 1-30);
>
> determining on the network interface device that said one CCB is stored in the DRAM and transferring said one CCB into the SRAM (e.g. Mayer, col. 31, lines 63-67; col. 32, lines 1-30); and
>
> transferring the data portion of the packet from the network interface device into a destination without transferring the header portion of the packet into the destination, the destination having been determined by the processing device (e.g. Mayer, col. 15, lines 46-52).
>
> Mayer fails to teach the method wherein the packet is a TCP/IP packet and wherein the network protocol stack executing on the processing device performs substantially no TCP protocol processing on the TCP/IP packet.
>
> However, in a similar art, Kapoor teaches a network communications device which operates on TCP/IP packets using a network protocol stack (e.g. Kapoor, col. 3, lines 49-51) and also the ability to avoid, or bypass, the processing of a packet on a TCP (transport)

> layer of the protocol stack (e.g. Kapoor, col. 2, lines 33-49; col. 6, lines 7-19; Fig. 2B).
>
> It would have been obvious to one skilled in the art at the time the invention was made to combine Kapoor with Mayer because of the advantages of allowing a packet to avoid processing in various layers of a protocol stack. Kapoor teaches that bypassing the transport layer "provides significant performance gains" for the transmission of packets through the network (e.g. Kapoor, col. 2, lines 44-49). These performance gains include increases in both speed and efficiency of packet transmission, since unnecessary processing is avoided and the packet can be transferred directly to its intended destination. The processor on the network communication device performs the processing that is needed on the packet, thereby allowing the central processing unit of the host, or main computer, to perform other tasks, further increasing the speed and efficiency of packet transmission, which is beneficial in any computer network system.

Applicants respectfully disagree with the Office Action assertion that Mayer teaches "maintaining a set of communication control blocks (CCBs), some of the set of CCBs being maintained in a static random access memory (SRAM), others of the set of CCBs being maintained in a dynamic random access memory (DRAM)… (e.g. Mayer, several types of 'control blocks' are disclosed, high speed bus control block, memory control block, processor control block, all relating to communication procedures also the maintaining of the control blocks in both DRAM and SRAM, col. 17, lines 11-17; col. 31, lines 63-67; col. 32, lines 1-30)." Mayer instead discloses ***controller*** blocks that cannot be "maintained ***in*** a static random access memory (SRAM)" or "maintained ***in*** a dynamic random access memory (DRAM)" because they include controller hardware such as interfaces, registers and FIFOs. See, e.g., FIG. 4 of Mayer.

Applicants also respectfully disagree with the Office Action assertion that Mayer teaches "using content addressable memory (CAM) on the network interface device to determine that the packet is associated with one of the first plurality of CCBs (e.g. Mayer, CAM is taught as an alternative form of SRAM, similar in both use and function, col. 2, lines 28-35; col. 32, lines 1-30)." Applicants respectfully assert that Mayer does not teach "using content addressable memory (CAM) on the network interface device to determine that the packet is associated with one of the first plurality of CCBs," in the referenced sections or elsewhere.

Applicants further respectfully disagree with the Office Action assertion that Mayer teaches "determining on the network interface device that said one CCB is stored in the DRAM and transferring said one CCB into the SRAM (e.g. Mayer, col. 31, lines 63-67; col. 32, lines 1-30)." Column 31, line 63 – column 32, line 30 of Mayer are reprinted below:

> An RX HCB interface 601 is coupled to the bus 420 including the MDO[3 1:0] signals, and includes a data output (DOUT) for providing data to a first multi-line input of a four-input data mux 632 across a bus 620, where the mux 632 provides its output to MemDataOut inputs of the DRAM controller 636. The RX HCB interface 601 includes STB/CTL inputs for receiving the strobe and control signals of the DRAM RQ/GT/STB/CTL signals 628. An RX controller 604 is coupled to the bus 420, and has AD/LN/ST outputs coupled across a bus 612 to the second input of the mux 630. The RX controller 604 has a data output DOUT coupled to the second input of the mux 632 across a bus 622, a data input DIN coupled to the bus 618, SRAM RQ/GT/STB/CTL inputs for receiving SRAM RQ/GT/STB/CTL signals 654 associated with a static RAM (SRAM) 650 and DRAM RQ/GT/STB/CTL inputs for receiving the DRAM RQ/GT/STB/CTL signals 628.
>
> A TX HCB interface 605 is coupled to the bus 420 including the MDI[31:0] signals, and has a data input DIN coupled to the bus 618 and STB/CTL inputs receiving the strobe and control signals of the DRAM RQ/GT/STB/CTL signals 628. A TX controller 606 is coupled to the bus 420 and has AD/LN/ST outputs provided to the third input of the mux 630 across a bus 614, a data output DOUT coupled to the third input of the mux 632 across a bus 624, a data input DIN coupled to the bus 618, SRAM RQ/GT/STB/CTL inputs for receiving the SRAM RQ/GT/STB/CTL signals 654 and DRAM RQ/GT/STB/CTL inputs for receiving the DRAM RQ/GT/STB/CTL signals 628. The PCB interface 424 has AD/LN/ST outputs coupled to the fourth input of the mux 630 across a bus 616, a data output DOUT coupled to the fourth input of the mux 632 across a bus 626, a data input DIN coupled to the bus 618, SRAM RQ/GT/STB/CTL inputs for receiving the SRAM RQ/GT/STB/CTL signals 654 and DRAM RQ/GT/STB/CTL inputs for receiving the DRAM RQ/GT/STB/CTL signals 628.

These paragraphs do not teach "determining on the network interface device that said one CCB is stored in the DRAM," or "transferring said one CCB into the SRAM."

Applicants also respectfully disagree with the Office Action assertion that Mayer teaches "transferring the data portion of the packet from the network interface device into a destination without transferring the header portion of the packet into the destination, the

destination having been determined by the processing device (e.g. Mayer, col. 15, lines 46-52)." Instead, column 15, lines 46-52 of Mayer state:

> In the following cycles of the CLK signal, packet data is concurrently transferred or read from the source port and directly written to the destination port across the HSB 206 without being stored in the EPSM 210 or the memory 212. Data transfer occurs in cycles 5, 6, 7 and 8, for transferring several bytes depending upon the embodiment. For example, up to 64 bytes are transferred for L64381 devices, and up to 256 bytes are transferred for QE110 devices. Although four CLK cycles are shown for the data transfer, the data transfer may occur with one, two or four CLK cycles depending upon how much data is transferred. For new packets, a normal read cycle is first performed to provide the source and destination MAC addresses into the EPSM 210, which then performs a hashing procedure, described further below, to determine the destination port number, if known. Once the destination port number is known, and if there is only one destination port; a concurrent read and write operation may be performed for any portion or the entire remainder of the packet as desired.

Applicants respectfully assert that this paragraph does not teach "transferring the data portion of the packet from the network interface device into a destination without transferring the header portion of the packet."

For at least these reasons, applicants respectfully assert that Mayer does not teach or suggest many of the limitations of claim 1, in contrast to the Office Action assertions.

In addition, applicants agree with the Office Action statement that "Mayer fails to teach the method wherein the packet is a TCP/IP packet and wherein the network protocol stack executing on the processing device performs substantially no TCP protocol processing on the TCP/IP packet."

Applicants respectfully disagree, however, with the Office Action assertion that "Kapoor teaches a network communications device which operates on TCP/IP packets using a network protocol stack (e.g. Kapoor, col. 3, lines 49-51) and also the ability to avoid, or bypass, the processing of a packet on a TCP (transport) layer of the protocol stack (e.g. Kapoor, col. 2, lines 33-49; col. 6, lines 7-19; Fig. 2B)."

Applicants note that Kapoor is instead directed to "managing remote procedure calls (RPC's) ***between client and server processes running on the same host computer*** in a network." Kapoor, column 2, lines 34-36, emphasis added. Similarly, Kapoor teaches: "It is another object of the invention to enable a network RPC mechanism to ***distinguish***

whether client and server processes are running on the same host machine and, ***if so, to bypass the network and transport layers*** using an alternate protocol sequence that exploits local interprocess communication (IPC) facilities." Kapoor, column 2, lines 37-43, emphasis added. Likewise, Kapoor teaches: "It is a more specific object of the invention to ***recognize when client and server DCE processes exist on the same host to thereby use a local IPC mechanism, instead of a network layer (IP) or transport layer (TP) path***, to implement a remote procedure call." Kapoor, column 2, lines 44-48, emphasis added. In contrast to these teachings of Kapoor, claim 1 recites in part "receiving a TCP/IP packet onto the network interface device from a network...and transferring the data portion of the TCP/IP packet from the network interface device and into a destination...wherein the network protocol stack executing on the processing device performs substantially no TCP protocol processing on the TCP/IP packet." Kapoor does not teach or suggest these limitations because the client and server processes of Kapoor are running on the same host computer.

Moreover, applicants respectfully disagree with the Office Action assertion that "Kapoor teaches that bypassing the transport layer "provides significant performance gains" for the transmission of packets through the network (e.g. Kapoor, col. 2, lines 44-49)." As noted above, the performance gains alleged by Kapoor are not over a network but rather when the "client and server processes are running on the same host machine."

In addition, applicants respectfully disagree with the Office Action assertion that "The processor on the network communication device performs the processing that is needed on the packet, thereby allowing the central processing unit of the host, or main computer, to perform other tasks." Instead, Kapoor states: "VISA, our Virtual Internet SCSI Adapter ...is the OS mechanism for supporting access to storage peripherals via the network." Kapoor, page 72, column 1, lines 27-29. Thus, applicants respectfully but strongly disagree with the Office Action allegation that "It would have been obvious to one skilled in the art at the time the invention was made to combine Kapoor with Mayer because of the advantages of allowing a packet to avoid processing in various layers of a protocol stack."

For all the above reasons, applicants respectfully assert that claim 1 and all the claims that depend from claim 1 are not obvious over the cited references.

Regarding independent claim 14, the Office Action states:

As per claim 14, Mayer teaches a network interface device that is coupled to a processing device, the processing device executing a protocol stack, the network interface device comprising:

an amount of SRAM, the SRAM storing a first plurality of communication control blocks (CCBs) that are under control of the network interface device (e.g. Mayer, col. 17, lines 11-17; col. 31, lines 63-67; col. 32, lines 1-30);

an amount of DRAM, the DRAM storing a second plurality of CCBs that are under control of the network interface device (e.g. Mayer, col. 17, lines 11-17; col. 31, lines 63-67; col. 32, lines 1-30);

specialized hardware that analyzes a packet received onto the network interface device from a network, the packet comprising a data portion and a header portion, the specialized hardware generating a summary from the packet (e.g. Mayer, col. 7, lines 52-67; col. 3, lines 21-30; col. 19, lines 53-67);

a processor that uses the summary and a content addressable memory (e.g. Mayer, col. 2, lines 28-35; col. 58, lines 44-51), the TCP/IP packet being associated with one of the second plurality of CCBs, the processor causing said one of the second plurality of CCBs to be moved from the DRAM into the SRAM (e.g. Mayer, col. 31, lines 63-67; col. 32, lines 1-30); and

a mechanism that moves the data portion of the packet from the network interface device and into a destination identified by the processing device, the data portion of the packet being written into the destination without the header portion of the packet being written into the destination (e.g. Mayer, col. 15, lines 46-52).

Mayer fails to teach the network interface device wherein the packet is a TCP/IP packet and using the summary to determine whether the TCP/IP packet can be processed via a fast-path by the network interface device as opposed to being processed via a slow-path using the protocol stack, the data portion of the TCP/IP packet being written into the destination without the header portion of the TCP/IP packet being written into the destination and without the protocol stack doing substantial TCP protocol processing on the TCP/IP packet.

However, in a similar art, Kapoor teaches a network communications device which operates on TCP/IP packets using a network protocol stack (e.g. Kapoor, col. 3, lines 49-51) and also the ability to avoid, or bypass, the processing of a packet on a TCP (transport) layer of the protocol stack (e.g. Kapoor, col. 2, lines 33-49; col. 6, lines 7-19; Fig. 2B; col. 5, lines 36-45).

It would have been obvious to one skilled in the art at the time the invention was made to combine Kapoor with Mayer for similar reasons as stated above in regards to claim 1.

Applicants respectfully disagree with the Office Action assertion that Mayer teaches "an amount of SRAM, the SRAM storing a first plurality of communication control blocks (CCBs) that are under control of the network interface device (e.g. Mayer, col. 17, lines 11-17; col. 31, lines 63-67; col. 32, lines 1-30)." Mayer instead discloses ***controller*** blocks that cannot be "maintained ***in*** a static random access memory (SRAM)" because they include controller hardware such as interfaces, registers and FIFOs. See, e.g., FIG. 4 of Mayer.

Applicants also respectfully disagree with the Office Action assertion that Mayer teaches "an amount of DRAM, the DRAM storing a second plurality of CCBs that are under control of the network interface device (e.g. Mayer, col. 17, lines 11-17; col. 31, lines 63-67; col. 32, lines 1-30)." Mayer instead discloses ***controller*** blocks that cannot be "maintained ***in*** a dynamic random access memory (DRAM)" because they include controller hardware such as interfaces, registers and FIFOs. See, e.g., FIG. 4 of Mayer.

Moreover, applicants note that the Office Action admits that neither Mayer, Kapoor or the Office Action's proposed combination of Mayer and Kapoor teach "a processor that uses the summary and a content addressable memory to determine whether the TCP/IP packet can be processed via a fast-path by the network interface device as opposed to being processed via a slow-path using the protocol stack," in contrast to claim 14.

In addition, applicants agree with the Office Action statement that "Mayer fails to teach the network interface device wherein the packet is a TCP/IP packet and using the summary to determine whether the TCP/IP packet can be processed via a fast-path by the network interface device as opposed to being processed via a slow-path using the protocol stack, the data portion of the TCP/IP packet being written into the destination without the header portion of the TCP/IP packet being written into the destination and without the protocol stack doing substantial TCP protocol processing on the TCP/IP packet."

Applicants respectfully disagree, however, with the Office Action assertion that "Kapoor teaches a network communications device which operates on TCP/IP packets using a network protocol stack (e.g. Kapoor, col. 3, lines 49-51) and also the ability to

avoid, or bypass, the processing of a packet on a TCP (transport) layer of the protocol stack (e.g. Kapoor, col. 2, lines 33-49; col. 6, lines 7-19; Fig. 2B)."

Applicants note that Kapoor is instead directed to "managing remote procedure calls (RPC's) ***between client and server processes running on the same host computer*** in a network." Kapoor, column 2, lines 34-36, emphasis added. Similarly, Kapoor teaches: "It is another object of the invention to enable a network RPC mechanism to ***distinguish whether client and server processes are running on the same host machine*** and, ***if so, to bypass the network and transport layers*** using an alternate protocol sequence that exploits local interprocess communication (IPC) facilities." Kapoor, column 2, lines 37-43, emphasis added. Likewise, Kapoor teaches: "It is a more specific object of the invention to ***recognize when client and server DCE processes exist on the same host to thereby use a local IPC mechanism, instead of a network layer (IP) or transport layer (TP) path***, to implement a remote procedure call." Kapoor, column 2, lines 44-48, emphasis added. In contrast to these teachings of Kapoor, claim 14 recites in part "specialized hardware that analyzes a TCP/IP packet received onto the network interface device from a network...and a mechanism that moves the data portion of the TCP/IP packet from the network interface device and into a destination identified by the processing device,." Kapoor does not teach or suggest these limitations because the client and server processes of Kapoor are running on the same host computer.

Moreover, applicants respectfully but strongly disagree with the Office Action allegation that "It would have been obvious to one skilled in the art at the time the invention was made to combine Kapoor with Mayer for similar reasons as stated above in regards to claim 1." Applicants respectfully assert that, as discussed above, one of such skill would not have made the combination that the Office Action proposes.

Assuming arguendo that one of skill in the art would have made the combination of Mayer and Kapoor proposed by the Office Action, applicants respectfully assert that such a combination would not have "specialized hardware that analyzes a (TCP/IP) packet received onto the network interface device from a network, the (TCP/IP) packet comprising a data portion and a header portion, the specialized hardware generating a summary from the (TCP/IP) packet (e.g. Mayer, col. 7, lines 52-67; col. 3, lines 21-30; col. 19, lines 53-67)," in contrast to claim 14. Applicants respectfully assert that TCP/IP

hardware is not trivial and is a far cry from the teaching in column 3, lines 24-25 of Mayer of "hash logic that receives and hashes the network addresses to determine the hash address," which is not a summary.

For all the above reasons, applicants respectfully assert that independent claim 14 and all the claims that depend from claim 14 are not obvious over the cited references.

Regarding independent claim 17, the Office Action states:

> As per claim 17, Mayer teaches a network interface device that is integrated with a processing device, the processing device executing a protocol stack, the network interface device comprising:
>
> an amount of SRAM, the SRAM storing a first plurality of communication control blocks (CCBs) that are under control of the network interface device (e.g. Mayer, col. 17, lines 11-17; col. 31, lines 63-67; col. 32, lines 1-30);
>
> an amount of DRAM, the DRAM storing a second plurality of CCBs that are under control of the network interface device (e.g. Mayer, col. 17, lines 11-17; col. 31, lines 63-67; col. 32, lines 1-30);
>
> means for analyzing a packet received onto the network interface device from a network and for generating from the packet a summary, the packet comprising a data portion and a header portion, the specialized hardware generating a summary from the packet (e.g. Mayer, col. 7, lines 52-67; col. 3, lines 21-30; col. 19, lines 53-67), the header portion including a destination port value and a source port value (e.g. Mayer, col. 58, lines 35-44);
>
> a processor that uses the summary and a content addressable memory, wherein the packet is associated with a second plurality of CCBs, the processor causing said one of the second plurality of CCBs to be moved from the DRAM into the SRAM (e.g. Mayer, col. 31, lines 63-67; col. 32, lines 1-30); and
>
> a mechanism that moves the data portion of the packet from the network interface device and into a destination accessible by the processing device, the data portion of the packet being written into the destination without the header portion of the packet being written into the destination (e.g. Mayer, col. 15, lines 46-52).
>
> Mayer fails to teach the network interface device that uses the summary and a content addressable memory to determine whether the packet can be processed via a fast-path by the network interface device as opposed to being processed via a slow-path using the protocol stack, the data portion of the packet being written without the protocol stack doing substantial TCP protocol processing on the packet.
>
> However, in a similar art, Kapoor teaches a network communications device which operates on TCP/IP packets using a network protocol stack (e.g. Kapoor, col. 3, lines 49-51) and also the ability to avoid, or bypass, the processing of a packet on a TCP (transport)

> layer of the protocol stack (e.g. Kapoor, col. 2, lines 33-49; col. 6, lines 7-19; Fig. 2B; col. 5, lines 36-45).
>
> It would have been obvious to one skilled in the art at the time the invention was made to combine Kapoor with Mayer for similar reasons as stated above in regards to claim 1.

Applicants respectfully disagree with the Office Action assertion that Mayer teaches "an amount of SRAM, the SRAM storing a first plurality of communication control blocks (CCBs) that are under control of the network interface device (e.g. Mayer, col. 17, lines 11-17; col. 31, lines 63-67; col. 32, lines 1-30)." Mayer instead discloses ***controller*** blocks that cannot be "maintained ***in*** a static random access memory (SRAM)" because they include controller hardware such as interfaces, registers and FIFOs. See, e.g., FIG. 4 of Mayer.

Applicants also respectfully disagree with the Office Action assertion that Mayer teaches "an amount of DRAM, the DRAM storing a second plurality of CCBs that are under control of the network interface device (e.g. Mayer, col. 17, lines 11-17; col. 31, lines 63-67; col. 32, lines 1-30)." Mayer instead discloses ***controller*** blocks that cannot be "maintained ***in*** a dynamic random access memory (DRAM)" because they include controller hardware such as interfaces, registers and FIFOs. See, e.g., FIG. 4 of Mayer.

Moreover, applicants note that the Office Action assertion that Mayer teaches "the header portion including a destination port value and a source port value (e.g. Mayer, col. 58, lines 35-44)" ignores the fact that Mayer's "ports 104, 110" are physical couplers (see Mayer, column 6, line 50 – column 7, line 30; FIG. 2), in contrast to the logical "TCP destination port value and a TCP source port value" recited in claim 17.

Applicants also respectfully disagree with the Office Action assertion that Mayer teaches "a mechanism that moves the data portion of the packet from the network interface device and into a destination accessible by the processing device, the data portion of the packet being written into the destination without the header portion of the packet being written into the destination (e.g. Mayer, col. 15, lines 46-52)." Instead, column 15, lines 46-52 of Mayer state:

> In the following cycles of the CLK signal, packet data is concurrently transferred or read from the source port and directly written to the destination port across the HSB 206 without being stored in the EPSM 210 or the memory 212. Data transfer occurs in cycles 5, 6, 7 and

> 8, for transferring several bytes depending upon the embodiment. For example, up to 64 bytes are transferred for L64381 devices, and up to 256 bytes are transferred for QE110 devices. Although four CLK cycles are shown for the data transfer, the data transfer may occur with one, two or four CLK cycles depending upon how much data is transferred. For new packets, a normal read cycle is first performed to provide the source and destination MAC addresses into the EPSM 210, which then performs a hashing procedure, described further below, to determine the destination port number, if known. Once the destination port number is known, and if there is only one destination port; a concurrent read and write operation may be performed for any portion or the entire remainder of the packet as desired.

Applicants respectfully assert that this paragraph does not teach "“a mechanism that moves the data portion of the packet from the network interface device and into a destination accessible by the processing device, the data portion of the packet being written into the destination without the header portion of the packet being written into the destination."

Moreover, applicants note that the Office Action admits that neither Mayer, Kapoor or the Office Action's proposed combination of Mayer and Kapoor teach "a processor that uses ... a content addressable memory to determine whether the packet can be processed via a fast-path by the network interface device as opposed to being processed via a slow-path using the protocol stack," in contrast to claim 17.

For at least these reasons, applicants respectfully assert that Mayer does not teach or suggest many of the limitations of claim 17, in contrast to the Office Action assertions.

In addition, applicants agree with the Office Action statement that "Mayer fails to teach the network interface device that uses the summary and a content addressable memory to determine whether the packet can be processed via a fast-path by the network interface device as opposed to being processed via a slow-path using the protocol stack, the data portion of the packet being written without the protocol stack doing substantial TCP protocol processing on the packet."

Applicants respectfully disagree, however, with the Office Action assertion that "Kapoor teaches a network communications device which operates on TCP/IP packets using a network protocol stack (e.g. Kapoor, col. 3, lines 49-51) and also the ability to

avoid, or bypass, the processing of a packet on a TCP (transport) layer of the protocol stack (e.g. Kapoor, col. 2, lines 33-49; col. 6, lines 7-19; Fig. 2B)."

Applicants note that Kapoor is instead directed to "managing remote procedure calls (RPC's) ***between client and server processes running on the same host computer*** in a network." Kapoor, column 2, lines 34-36, emphasis added. Similarly, Kapoor teaches: "It is another object of the invention to enable a network RPC mechanism to ***distinguish whether client and server processes are running on the same host machine*** and, ***if so, to bypass the network and transport layers*** using an alternate protocol sequence that exploits local interprocess communication (IPC) facilities." Kapoor, column 2, lines 37-43, emphasis added. Likewise, Kapoor teaches: "It is a more specific object of the invention to ***recognize when client and server DCE processes exist on the same host to thereby use a local IPC mechanism, instead of a network layer (IP) or transport layer (TP) path***, to implement a remote procedure call." Kapoor, column 2, lines 44-48, emphasis added. In contrast to these teachings of Kapoor, claim 14 recites in part "specialized hardware that analyzes a TCP/IP packet received onto the network interface device from a network...and a mechanism that moves the data portion of the TCP/IP packet from the network interface device and into a destination identified by the processing device." Kapoor does not teach or suggest these limitations because the client and server processes of Kapoor are running on the same host computer.

Moreover, applicants respectfully but strongly disagree with the Office Action allegation that "It would have been obvious to one skilled in the art at the time the invention was made to combine Kapoor with Mayer for similar reasons as stated above in regards to claim 1." Applicants respectfully assert that, as discussed above, one of such skill would not have made the combination that the Office Action proposes.

Assuming arguendo that one of skill in the art would have made the combination of Mayer and Kapoor proposed by the Office Action, applicants respectfully assert that such a combination would not have "specialized hardware that analyzes a (TCP/IP) packet received onto the network interface device from a network, the (TCP/IP) packet comprising a data portion and a header portion, the specialized hardware generating a summary from the (TCP/IP) packet (e.g. Mayer, col. 7, lines 52-67; col. 3, lines 21-30; col. 19, lines 53-67)," in contrast to claim 17. Applicants respectfully assert that TCP/IP

hardware is not trivial and is a far cry from the teaching in column 3, lines 24-25 of Mayer of "hash logic that receives and hashes the network addresses to determine the hash address," which is not a summary.

For all the above reasons, applicants respectfully assert that claim 17 and all the claims that depend from claim 17 are not obvious over the cited references.

B. The Office Action rejects claim 7 under 35 U.S.C. §103(a) as being unpatentable over Mayer in view of Kapoor and further in view of U.S. Patent No. 5,991,299 to Radogna et al. (hereinafter "Radogna"). Regarding claim 1, the Office Action states:

> As per claim 7, Mayer and Kapoor teach the method of claim 6, but fail to teach the method wherein the specialized hardware comprises a sequencer.
>
> However, in a similar art, Radogna teaches a network communication system that is able to accelerate packet header protocol processing, including the use of a sequencer (e.g. Radogna, col. 2, lines 15-28).
>
> It would have been obvious to one skilled in the art at the time the invention was made to combine Radogna with Mayer and Kapoor because of the advantages of using a sequencer to perform specific network processing tasks. Additionally, specialized hardware, such as a sequencer to assist the central processing unit in handling simple, redundant tasks that need to be completed on every incoming and/or outgoing packet packet. This can greatly reduce processing time since the CPU is not overloaded with menial tasks and is available to handle more complex processes faster and more efficiently. Processing and transferring a packet through a network faster and more efficiently is beneficial in any computer network.

Applicants respectfully disagree with the Office Action proposition that "It would have been obvious to one skilled in the art at the time the invention was made to combine Radogna with Mayer and Kapoor because of the advantages of using a sequencer to perform specific network processing tasks." Specifically, applicants respectfully disagree with the Office Action assertion that "specialized hardware, such as a sequencer to assist the central processing unit in handling simple, redundant tasks that need to be completed on every incoming and/or outgoing packet." Because this assertion appears to come from the Examiner's personal knowledge rather than from any teaching in the cited

references, applicants respectfully request the Examiner to provide a supporting affidavit, as required by 37 C.F.R. §1.104(d)(2). Additionally, applicants respectfully disagree with the Office Action assertion that "This can greatly reduce processing time since the CPU is not overloaded with menial tasks and is available to handle more complex processes faster and more efficiently." Because this assertion also appears to come from the Examiner's personal knowledge rather than from any teaching in the cited references, applicants respectfully request the Examiner to provide a supporting affidavit, as required by 37 C.F.R. §1.104(d)(2). Moreover, applicants respectfully disagree with the conclusory Office Action assertion that "Processing and transferring a packet through a network faster and more efficiently is beneficial in any computer network," and applicants respectfully assert that there are situations for which processing a packet more slowly makes more sense.

For all the above reasons, applicants respectfully assert that claim 7 is not obvious over the cited references.

II. Conclusion

Applicants have responded to each of the items in the Office Action, and believe that all of the pending claims are in condition for allowance. As such, applicants respectfully solicit a Notice of Allowance.

Respectfully submitted,

Mark Lauer
Reg. No. 36,578
6601 Koll Center Parkway
Suite 245
Pleasanton, CA 94566
Tel: (925) 484-9295
Fax: (925) 484-9291

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the U.S. Postal Service with sufficient postage as first class mail in an envelope addressed to: the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on May 8, 2006.

Date: 5-8-06

Mark Lauer